

MENU

SEARCH

INDEX

E4910

1/1



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11)Publication number: 09223054

(43)Date of publication of application: 26.08.1997

(51)Int.Cl.

G06F 12/00

G06F 9/06

(21)Application number: 08027105

(71)Applicant:

NIPPON STEEL CORP  
SHINNITTETSU JOHO TSUSHIN  
SYST KK

(22)Date of filing: 14.02.1996

(72)Inventor:

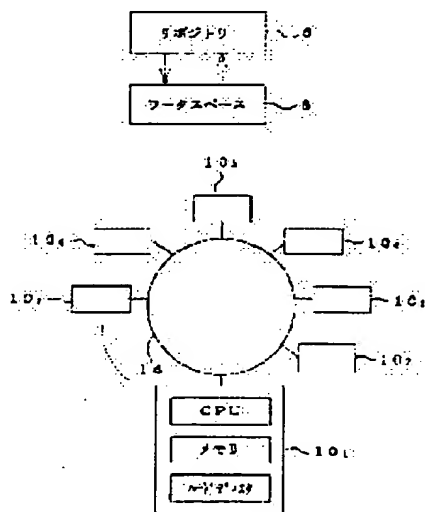
MATSUZAKA YASUHIRO  
UCHIDA YASUO

(54) FILE SET MANAGEMENT DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To reduce the burden or worker by handling the files in every set of plural files and also performing the version management in every file set.

SOLUTION: The hosts 101, 102... are connected to each other via a communication circuit 14 in a computer network. The file set management device of these hosts consists of a repository 6 serving as a file storage means and a work space 8. In the space 8, every worker performs the production, updating, etc., of programs. Then, the repository 6 stores the file of programs produced in the space 8. The files are read out of the repository 6 and also stored there in every set of files. At the same time, the version management is carried out for every set of files.



LEGAL STATUS

**This Page Blank (uspto)**

(19)日本国特許庁(JP)

(12) 公開特許公報(A)

(11)特許出願公開番号

特開平9-223054

(43)公開日 平成9年(1997)8月26日

(51)Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	FI	技術表示箇所
G 0 6 F 12/00	5 1 7		G 0 6 F 12/00	5 1 7
9/06	4 1 0		9/06	4 1 0 P

審査請求 未請求 請求項の数4 OL (全 6 頁)

(21)出願番号 特願平8-27105

(22)出願日 平成8年(1996)2月14日

(71)出願人 000006655

新日本製鐵株式会社

東京都千代田区大手町2丁目6番3号

(71)出願人 000191076

新日鉄情報通信システム株式会社

東京都中央区新川2丁目20番15号

(72)発明者 松坂 泰洋

東京都千代田区大手町2丁目6番3号 新

日本製鐵株式会社内

(72)発明者 内田 康夫

東京都中央区新川2丁目20番15号 新日鉄

情報通信システム株式会社内

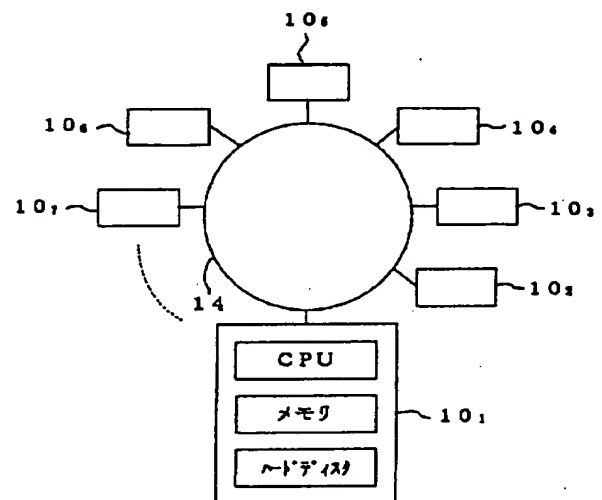
(74)代理人 弁理士 半田 昌男

(54)【発明の名称】 ファイル集合管理装置

(57)【要約】

【課題】 作業者が効率よく必要なファイルを取り扱うことができるファイル集合管理装置を提供する。

【解決手段】 プログラムの構成要素であって、特定の機能又は手続を実現する所定の複数のファイルを考え、この複数のファイル及びこれに関連した必要なテキストファイルの一つのまとまりとして、バージョン管理やその他の処理を行う。このような複数のファイルを「部品構造体」と呼ぶ。部品構造体は、コンパイラによって要求されるディレクトリ構造とは関係なく、したがって、コンパイルするファイルが異なるディレクトリに属していても、作業者はそのことを意識せずに部品構造体単位でリポジトリからワークスペースへのファイルの読み出し及びワークスペースからリポジトリへのファイルの格納を行うことができる。また、部品構造体単位でバージョン管理を行う。



【特許請求の範囲】

【請求項1】 複数のファイルからなるファイル集合を単位として、記憶手段からのファイルの読み出し及び前記記憶手段に対するファイルの格納を行うとともに、前記ファイル集合を単位としてバージョン管理を行うことを特徴とするファイル集合管理装置。

【請求項2】 前記ファイル集合は、コンピュータプログラムの一部を構成するファイルを含むものであることを特徴とする請求項1記載のファイル集合管理装置。

【請求項3】 前記ファイル集合は、ドキュメントファイルを含むものであることを特徴とする請求項1記載のファイル集合管理装置。

【請求項4】 前記ファイル集合は、別のファイル集合をも含むものであることを特徴とする請求項1、2又は3記載のファイル集合管理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータプログラム等の構成要素であって、複数のファイルの集合からなるものを管理するファイル集合管理装置に関するものである。

【0002】

【従来の技術】規模の大きいコンピュータプログラムを開発する場合、そのプログラムを、特定の機能を有する多数の構成要素に分けて、複数の作業者が自分の担当する構成要素のプログラムを作成し、最後にこれらを合成して大きな一つのプログラムを完成させるという手法が採られることが多い。分割された各構成要素のプログラムは、一般に複数のファイルの集合からなる。このようなプログラムの構成要素に、その構成要素の機能等を記載したテキストファイルなどを付随させたファイルの集合を部品と呼ぶ。この部品は、特定の機能に関連しているため、作業者は、ファイル単位よりも、むしろ部品単位でそのプログラムを考える。このため、作業者がファイルを取り扱う場合、この部品を構成するファイルの集合を単位として取り扱うことができれば、便利である。

【0003】このような複数のファイルからなる部品を単位として取り扱う方法として、それぞれのファイルに対し、それが属する部品を識別するためのタグをつける方法がある。これによって、作業者はそのタグを見れば、そのファイルがどの部品に属しているかを知ることができる。また、例えばある部品に属するすべてのファイルが必要となったときには、そのタグをパラメータとして読み出しの指示を行うことによって、簡単に、その部品に属するファイルだけをリポジトリからワークスペース上に取り出すことができる。

【0004】

【発明が解決しようとする課題】ところで、全体のプログラムを構成するファイルの数が数千から数万に及ぶような場合には、部品の数も多くなり、タグ名を見ただけ

ではその部品がどのようなものを直ちに理解するのは難しい。また、複数の作業者がリポジトリにある部品を共有する場合、ある作業者がリポジトリから必要な部品をワークスペース上に取り出して改良し、それを再びリポジトリに登録するときには、過去のバージョンと最新のバージョンとの関係について、他の作業者也把握しておかなければならない。更に、ファイルだけでなく別の部品を組み合わせて新しい一つの部品を構成するという場合のように、部品同士の間に関係がある場合がある。

【0005】従来、多数のファイルや部品を自動的に効率よく管理するツールはなく、作業者が、専用の台帳を用意し、タグ名と部品の機能や内容などとの関係、あるいは部品の階層関係などをこの台帳に記載するという方法で、人間の手によって各ファイルや部品を管理していた。本発明は上記事情に基づいてなされたものであり、作業者が効率よく必要なファイルや部品を取り扱うことができるファイル集合管理装置を提供することを目的とするものである。

【0006】

【課題を解決するための手段】上記の課題を解決するための請求項1記載の発明は、複数のファイルからなるファイル集合を単位として、記憶手段からのファイルの読み出し及び前記記憶手段に対するファイルの格納を行うとともに、前記ファイル集合を単位としてバージョン管理を行うことを特徴とするものである。

【0007】請求項2記載の発明は、請求項1記載の発明において、前記ファイル集合が、コンピュータプログラムの一部を構成するファイルを含むものであることを特徴とするものである。請求項3記載の発明は、請求項1記載の発明において、前記ファイル集合が、ドキュメントファイルを含むものであることを特徴とするものである。

【0008】請求項4記載の発明は、請求項1、2又は3記載の発明において、前記ファイル集合が、別のファイル集合をも含むものであることを特徴とするものである。

【0009】

【作用】たとえば、多数の作業者が分担して一つのプログラムを開発する場合は、各作業者に、そのプログラムの一部であって何らかのまとまった機能もしくは手続を実現できる構成要素を一つの単位として、開発を割り当てることが多い。プログラムが一つのまとまった機能もしくは手続を実現するためには、通常、種々のファイルが必要となり、これをコンパイラでオブジェクトコードに翻訳する場合には、それぞれのファイルを、そのコンパイラが要求するディレクトリに置くことが必要となる。このような場合に、ファイルの管理を個々のファイルごとに行うと、それぞれのファイルのディレクトリが異なることなどから、ファイルの管理が煩雑となる。ま

た、バージョンアップが繰り返されて複数のバージョン番号を有するファイルが存在する場合、そのファイルのどのバージョンを使用するかを各作業者が記憶しておくのは困難である。

【0010】そこで、作業者が実際の作業を行うときに取り扱う複数のファイルを、特定のファイル集合として定義して名称を付し、これを単位として取り扱うこととする。このファイル集合には、更に、別のファイル集合を含めることもできるようにする。一旦、このファイル集合を定義すれば、バージョン管理もこのファイル集合を単位として行うことができる。ファイルの記憶手段であるリポジトリからファイルを読み出すときは、ファイル集合の名称をパラメータとして、一回の操作で必要なファイルをすべて取り出すことができる。その際、たとえばその中のファイルが何回かバージョンアップされていて、そのファイル集合ではどのバージョンを使用していたかを作業者が忘れていたとしても、ファイル集合を単位として取り扱うこととすれば、問題なく必要なファイルを取り出すことができ、作業者の負担は軽減される。

【0011】

【発明の実施の形態】以下に図面を参照して、本発明の一実施形態を説明する。図1は、本実施形態のファイル集合管理装置の構成を示す図である。但し、これは、作業者が意識する概念上の構成であって、実際のハードウェア的な構成とは異なる。図1に示すように、本実施形態のファイル集合管理装置は、リポジトリ6とワークスペース8からなる。ワークスペース8は、各作業者がプログラムの作成、更新等の作業を行う領域であり、リポジトリ6は、ワークスペースで作成されたプログラムファイルなどが格納される領域である。尚、図1では、分かりやすくするためにワークスペース8を一つだけ示したが、実際には、並行して作業を進めている作業者の数だけワークスペース8は存在する。

【0012】図2は、本実施形態のファイル集合管理装置を適用するためのコンピュータ・ネットワークシステムの一例を示したブロック図である。同図に示すように、各ホスト10<sub>1</sub>、10<sub>2</sub>、・・・は、通信回線14を介して相互に接続されている。これらのホストは、例えばワークステーション等のコンピュータからなり、内部にメモリ、CPU、ハードディスク等を備えている。図1に示すリポジトリ6及びワークスペース8は、あくまでも論理的概念として存在するものであり、これらが存在するためのホストは特に決められていない。ワークスペース8において現在作成中もしくは修正中のファイル、並びにリポジトリ6に格納されているファイルは、物理的には、ファイル集合管理装置が指定する適当なホストのハードディスク上にある。作業者がリポジトリ6から特定のファイルを取り出すよう指示すると、そのファイルが格納されているホストのハードディスクがアクセスされてそのファイルが読みだされ、通信回線1

4を介してその作業者のワークスペース上に転送される。したがって、作業者は、そのファイルがどこに格納されているかを意識する必要はない。尚、本実施形態では、単一のファイルごとに読み出しや格納を行うのではなく、複数のファイルからなる部品構造体を単位として、読み出しや格納を行う。この「部品構造体」については後述する。

【0013】一般に、規模の大きいプログラムの開発を多数の作業者の協力のもとで行う場合、各作業には、そのプログラムの構成要素であって、特定の機能を実際に実行できるプログラムの開発が割り振られる。各作業者は、開発が割り振られた機能を実現できるプログラムを開発するために、その機能を、更に適当な複数の部分に分割し、分割されたそれぞれの部分を一つのファイルとしてプログラミング等を行う。こうして一人の開発担当者が作成した複数のプログラムファイルは、全体のプログラムの一部の機能を実現する構成要素となる。

【0014】このようにして作成された全体プログラムの各構成要素となるプログラムは、コンパイラによってそれぞれにオブジェクトコードに翻訳され、その後、各作業によって開発されたプログラムのオブジェクトコードをリンクすることによって、一つの機械語のコンピュータプログラムにまとめられる。一般的なコンパイラを使用して、それぞれの作業者が作成した各構成要素のソースコードをオブジェクトコードに翻訳する場合、ワークスペース上に存在するソースコードのファイルは、そのコンパイラによって決められた所定の形式で格納されていなければならない。しかし、この「所定の形式」とは、コンパイラがソースコードをオブジェクトコードに翻訳するのに都合のよいディレクトリの階層構造であって、作業者がプログラム開発を行う際に便利な形式ではなく、したがって、その作業者が開発している部分のプログラムを構成するそれぞれのファイルは、一般に異なるディレクトリに属している。このため、最小単位であるファイルごとにプログラムを管理するのは困難である。

【0015】したがって作業者は、ファイル単位で取り扱うよりも、一定の関係を有する複数のファイルをまとまりとして取り扱う方が都合がよい。この「複数のファイル」とは、例えば、全体のプログラムのうち自分が開発を担当する一部の構成要素のプログラムを構成する複数のプログラムファイルと、この部分のプログラムの機能説明などを記載したテキストファイルなどである。その際、各ファイルのファイル名と経路名を併せたものを、あたかもファイル名のようにみなせば、それぞれのファイルがどのディレクトリに属するものかを考える必要はない。

【0016】以上の観点から、本実施形態では、プログラムの構成要素であって、特定の機能又は手続を実現するための複数のファイル（その構成要素の規模が小さい

場合には単一のファイルでもよい)と、必要に応じてこのような構成要素の機能説明等を記載したドキュメントファイルを併せたものを「部品構造体」として定義し、この部品構造体を単位として、バージョン管理やその他の処理を行う。また、一つの部品構造体は、ファイルだけでなく、別の部品構造体を含むこともできるものとし、これをサブ部品構造体と呼ぶ。但し、部品構造体一つも含まない部品構造体や、部品構造体だけを含みファイル一つも含まない部品構造体も存在できる。

【0017】図3は、部品構造体の一例の構成を示した図である。一つの部品構造体には、「部品構造体の名前」、「部品構造体のバージョン番号」、「リポジトリへの格納情報」、「部品構造体が管理するファイルのファイル名」、「部品構造体が管理するサブ部品構造体のワークスペースにおけるID」が含まれる。「リポジトリへの格納情報」には、例えばその部品構造体を修正した後でリポジトリへ格納する場合に、その格納位置に関する情報が記載される。「部品構造体が管理するファイルのファイル名」には、そのファイルの経路名が付随する。部品構造体の実体は、実際にこれを構成する複数のファイルやサブ部品構造体であるが、部品構造体定義ファイルは、その部品構造体を定義するために必要な情報が記載された独立した一つのファイルである。このようにして定義された部品構造体は、リポジトリ6に登録される。リポジトリ6には、部品構造体を単位として、それに含まれる複数のファイルあるいはサブ部品構造体と一緒に格納されている。したがって、例えば同じファイルが複数の部品構造体に含まれる場合は、リポジトリでは、それぞれの部品構造体についてそのファイルを含んだ状態で格納されている。

【0018】一つの部品構造体を構成するファイルやサブ部品構造体はどのようなものであってもよく、特に制限はないが、一般には一人の作業者が取り扱う特定の機能を実現するプログラムに必要なファイルやサブ部品構造体を、一つの部品構造体としてまとめる。また、ファイルのバージョン管理も、この部品構造体を単位として行う。すなわち、その部品構造体に含まれるファイルを更新すると、その部品構造体のバージョン番号は所定の形式に従って自動的に更新される。

【0019】図4は、本実施形態のファイル管理装置において、リポジトリ6内に格納される部品構造体、およびワークスペース8における部品構造体の構成と実際にコンパイラにかけられるファイルのディレクトリとの関係を模式的に示した図である。同図において、リポジトリ6には、A、B、C、D、E、F、Gという7つの部品構造体が格納されている。各部品構造体からつながる「1.1」等の数字は、その部品構造体のバージョン番号を示す。各部品構造体には、その部品構造体を構成する複数のファイル(図示せず)が含まれており、また、部品構造体が別のサブ部品構造体を含むものもある。こ

の部品構造体同士の包含関係は、記号「sub」によって示される。具体的には、部品構造体Dの第1.1版は部品構造体Eのすべてのバージョンを含んでおり、部品構造体Gの第1.1版は部品構造体Eのすべてのバージョンを含んでいる。部品構造体によっては、バージョンアップによって複数のバージョン番号を有するものもある。リポジトリ6では、部品構造体単位でファイルが格納されるので、バージョンアップされる部品構造体に含まれるファイルについては、それぞれのバージョン番号の部品構造体ごとにそれぞれのファイルがリポジトリに存在することになる。各部品構造体は、リポジトリ6内において、例えば市販のデータ管理システムを用いて、部品構造体ごとにまとめられて、データ圧縮された形で格納されている。

【0020】作業者が、必要とする部品構造体を自己のワークスペース20に読み出す場合、その格納位置情報を入力して読み出しを指示すると、図4に示す構成管理ツール21が起動され、リポジトリにアクセスしてその部品構造体を読みだす。これによって、その部品構造体に含まれるファイル及びサブ部品構造体は、リポジトリ6から一括して読み出される。したがって、この場合も作業者は、必要なファイルやそのバージョン番号を特段意識することなく、部品構造体単位で作業を進めることができる。そして、読み出した部品構造体について保守、デバッグ等の作業を行い、作業が終了したら、再びリポジトリ6に格納するか、またはコンパイラ23にかけて翻訳を行う。このように、部品構造体単位で操作をすることができるようするために、本発明者は、新たに構成管理ツール21というプログラムを開発した。この構成管理ツールと部品構造体定義ファイルが一体となって、部品構造体単位でのファイルの取り扱いが可能となる。

【0021】図4のワークスペース8の右側に22<sub>1</sub>、22<sub>2</sub>、・・・で示した多数の四角形は、コンパイラ23によって翻訳されるファイルを示したものである。これらのファイルは、コンパイラ23が要求するディレクトリ構造になっていることを模式的に示したものである。各ファイル22<sub>1</sub>、22<sub>2</sub>、・・・は、必ずいずれかの部品構造体に含まれているが、同時に、コンパイラ23によって課される条件に適合するように、特定のディレクトリに属するよう指定されている。この指定は、そのファイルが属するディレクトリに至る経路名を、個々のファイルのファイル名に記載することによって行う。ファイルが作成される当初において、そのファイルの種類や機能から、コンパイラから見たときにそのファイルがどのディレクトリに属していなければならないかが予め分かる。したがって、作業者は、そのファイルを作成するときに、所定の経路名をファイル名に付しておく。このように経路名を指定しておくことによって、部品構造体単位でファイルを取り扱っても、最終的にコン

バイラが処理できるディレクトリの階層構造に従って各ファイルを並べることができる。しかし、作業者がリポジトリに対して各ファイルの格納や読み出しを行う段階では、そのファイルが関連するディレクトリについては全く意識する必要はなく、コンパイラ等のファイル処理ツールが要求するディレクトリの階層構造などを気にすることなく、自分が担当する作業に必要なファイルの全体をまとめて取り扱うことができる。

【0022】部品構造体単位で必要なプログラムの開発が終了したら、ワークスペース8上で、その部品構造体に含まれるファイルをコンパイラにかけて翻訳することができる。このようにして得られたオブジェクトコードのプログラム24は、必要に応じてリンカ(linker)等のツールを利用して、他の部品構造体を翻訳したものと結合して、最終的に一つの機械語プログラムとされる。

【0023】尚、本発明は上記の実施の形態に限定されるものではなく、その要旨の範囲内で種々の変更が可能である。例えば、上記実施形態では、それぞれがホストと呼ばれるワークステーションのネットワークに対して本発明を適用したが、これ以外にも、例えばサーバー及びクライアントからなるコンピュータネットワークシステムにも、同様に本発明を適用できる。

【0024】また、上記実施形態では、本発明のファイル集合が、一部のプログラムファイルとドキュメントファイルを併せた部品構造体である場合について説明したが、このファイル集合はプログラムファイルだけの集合であってもよいし、ドキュメントファイル（たとえばマニュアルが記載されたもの）だけの集合であってもよい。

【0025】

【発明の効果】以上説明したように、本発明によれば、

複数のファイルからなるファイル集合を単位としてファイルの取り扱いを行うことによって、作業者は、複数のファイルを単位としてバージョンアップやリポジトリへの格納、リポジトリからの読み出しを行えばよく、したがって、従来のように、ファイル集合の間に階層関係がある場合でも、人間が手作業でそれぞれのファイル集合同士の間の関係などを管理することが不要となり、作業者の負担が大幅に軽減されるファイル集合管理装置を提供することができる。

10 【図面の簡単な説明】

【図1】リポジトリとワークスペースとの関係を示した図である。

15 【図2】本実施形態のファイル集合管理装置を適用したコンピュータ・ネットワークシステムの一例を示したブロック図である。

【図3】部品構造体定義ファイルの一例の構成を示した図である。

【図4】リポジトリ内に格納される部品構造体、およびワークスペースにおける部品構造体の構成と実際にコンパイラにかけられるファイルのディレクトリとの関係を模式的に示した図である。

【符号の説明】

6 リポジトリ

8 ワークスペース

25 10<sub>1</sub>, 10<sub>2</sub>・・・ 端末

12 ホストコンピュータシステム

14 通信回線

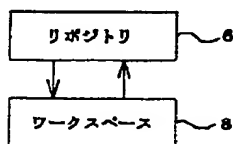
21 構成管理ツール

22<sub>1</sub>, 22<sub>2</sub>, ... ファイル

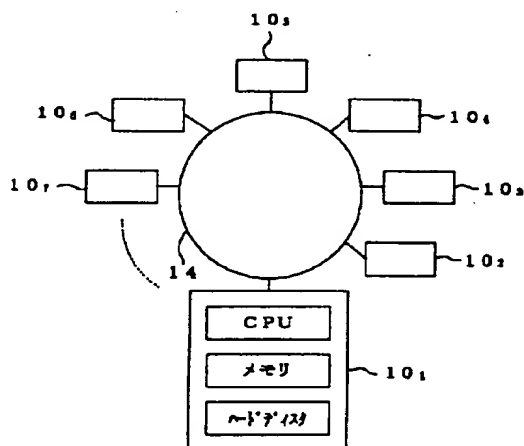
30 23 コンパイラ

24 プログラム

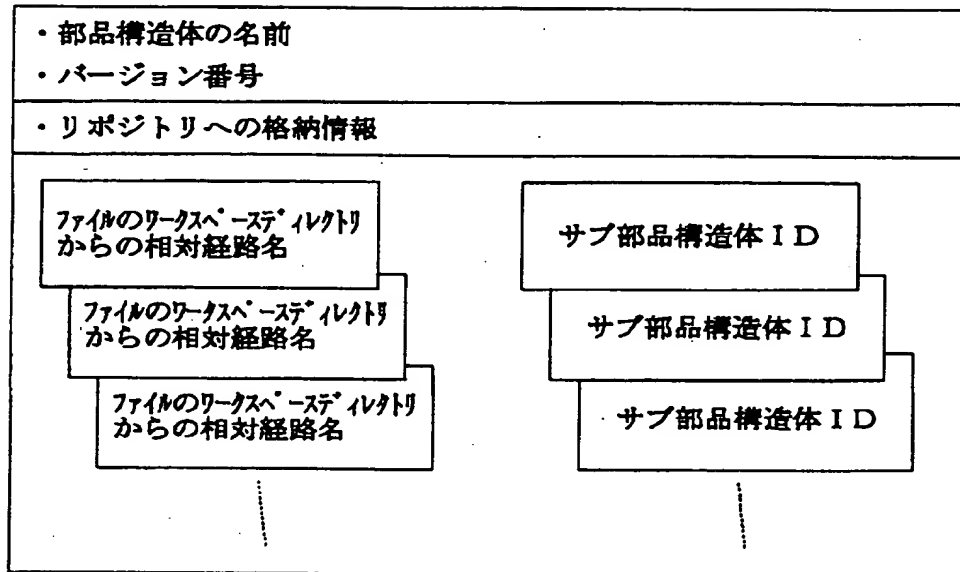
【図1】



【図2】



【図3】



【図4】

